Technische Universiteit
Eindhoven
University of Technology
Where innovation starts

UBET working on IBM compatible pc's : user's manual

Sniekers, R.J.J.M.

Published: 01/01/1990

Document Version
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Sniekers, R. J. J. M. (1990). UBET working on IBM compatible pc's : user's manual. (TH Eindhoven. Afd. Werktuigbouwkunde, Vakgroep Produktietechnologie : WPB; Vol. WPA0864). Eindhoven: Technische Universiteit Eindhoven.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 18. Dec. 2015

UBET working on IBM compatible PC's.

User's manual

R.J.J.M. Sniekers

March 90

VF code D1/D2/D3 WPA nr 0864

CONTENTS

	Contents	1
1	Introduction	2
2	UBET Installation	3
	2.1. Installing UBETTOOL	4
	2.2. Installing UBET programs	5
	2.3. Necessary changes to the PC system	5
3	Running Ubet	6
	3.1. Exit from UBET	8
4	Possible forging analysis	8
	4.1. Forging load prediction	8
	4.2. Flowlines prediction	8
	4.3. Advance die filling	9
	4.4. Die billet interfacial pressure	9
	Enclosure 1 Command's description	10
	Enclosure 2 Example of input	17
	Enclosure 3 Listing of batch files etc.	21
	3.1. UBETEXIS	21
	3.2. UBETSB	26

1. Introduction

Prof A.N. Bramley, Dr. F.A. Osman and their coworkers at the University of Bath developed the UBET program for VAX/VMS and SUN computers.
The PC version for UBET, which was developed at the Technical University of Denmark, was installed by Per Hanssen from the TUD during his stay in Bath.
After UBET was installed on the PC at the University of Bath several, people have been working on the UBET program in order to create new features for the program. Due to this fact there are at the moment several different source files for the current working versions of UBET.

In this paper the author wants to give an introduction of the use of UBET. In addition to this a short usermanual has been included.

2. UBET Installation

After a period of becoming familiar with the UBET program, the author's first task was to find out which of the existing source files still were necessary and which of them were out of date. This in order to find out how to compile and link the source files into an executable file.

After sorting out all the source files, it was possible to create several different programs. The differences between them involved for instance other implemented features or improved graphics. In order to separate them, each one is characterized with a name which is related to its contents.

PRELIMINARY UBET	: UBETPREL	:	the most preliminary PC UBET as developed at the TUD, and installed by Per Hansen.
ORIGINAL UBET	: UBETORIG	:	the updated version, main differences are about some changes in the graphic representation of the output.
EXISTING UBET	: UBETEXIS	:	closely related to the former, but including wear prediction programmes.
SONG BIN UBET	: UBETSB	:	in this program the optimisation routine is written by Song Bin instead of the optimization written by Dr. Osman.

For the last two versions of UBET, the author has made installation discs with an installation program, in order to :

1 separate the necessary sourcefiles.

2 compile and link the executable file.

3 provide the necessary datafiles.

4 separate source, object and executable files.

In enclosure 3 there is a list of the necessary files for UBETEXIS and UBETSB.In addition listings of the installation files are given.

Before installing UBET, UBETTOOL must be installed, which is a directory containing :

1 the necessary graphics drivers for running UBET, like INIT_VDI.

2 the compiler PROFORT.EXE and the linker LINK.EXE.

3 installation files for the different UBET programs.

4 the fortran libraries PROFORT, GKS and PFGKS.

2.1. Installing UBETTOOL

The root directory of the first UBETTOOL disc contains a file INSTTOOL.BAT. Copy this file to the rootdirectory of the harddisc, and activate INSTTOOL after changing the current drive and directory to C:\> . After this the installation file automatically performs the necessary commands. The only action from the operator is changing discs when the computer prompts for it.

2.2. Installing UBET program's

When the operator asks for a directory of UBETTOOL, there are two batchfiles for installing different UBET programs :

EXISINST.BAT

SBINST .BAT

Installing one of the UBET programs is simply done by activating the corresponding batchfile, and providing the discs the computer prompts for.

2.3. Necessary changes to the PC system

After the installation of UBETTOOL two system files have to be changed in order to get proper graphics

1 In the AUTOEXEC.BAT the following line has to be added:

C:\UBETTOOL\INIT_VDI

This line initializes the virtual device interface INIT_VDI.EXE.

2 In the CONFIG.SYS the following two lines have to be added:

DEVICE=C:\UBETTOOL\VDIDY010.SYS /R /G : DISPLAY

DEVICE=C:\UBETTOOL\VDI.SYS

The first line sets the Enhanced Color Display (EGA) Driver.

The second line sets the PC Virtual Device Driver.

3. Running UBET

After installation of a version (UBETEXIS or UBETSB) you have to change directory to the directory where that version is stored. The directory has the same name as the installed version.
For starting the ADVANCE mode (the EXE file is called ADV), type in ADV <> and in order to start the REVERSE mode (EXE file is called REV), type in REV <>.

After starting the program in the above way, the program will prompt

Plotter Connected.....Enter <Y> or <N>

Type in N <>, the program will prompt :

REFERENCE:

Anything you type will be accepted but it's very useful to type something that's traceable, for instance your name and a number or the part's name.
After this a prompt of

GO>

will appear on the screen.

After this you have to define the specific component and die geometries. This can be done in two different ways. First it's possible to read an existing file with component and die geometries, and second it's possible to input new component and die geometries.

Reading an existing file is possible with the command READ. Making new input can be done with the commands BILLET, BOTTOM and TOP for respectively billet, bottom die and top die coördinates. After making new input this can be saved by writing it into a datafile with the command WRITE.

The next stage, in order to let the program be able to perform calculations, will be the definition of the displacement of the top die by using the command PENETRATION, or its abbreviated form PEN, followed by the displacement in millimeters.

Besides these specific informations (geometries and displacement) UBET requires other parameters such as friction factor, process type, material properties and die velocity. Those four parameters are internally imbedded. By using the command DATA you can view the default data.

If any changes are necessary, you can change the friction factor by typing :

FRI followed by its new value, range is 0.0 to 1.0

The die velocity can be changed by :

VEL followed by its new value, in mm/sec

The stress relation can be changed by :

STRESS followed by A B ,A and B from the relation $\sigma = A + B \cdot \epsilon$

The process type can be changed by :

PRO followed by its new type, hot, cold or normal

If the process type is hot, then you have to set the following extra commands.

The material can be changed by :

MATERIAL followed by the chosen material, LCS, MCS, HCS or EN8

The process temperature can be changed by :

TEMPERATURE followed by its new value, in degrees Celsius.

Enclosure 1 contains a survey of the commands used in the UBET programs. Enclosure 2 lists the input for the analysis of one specific forging.

3.1. Exit from UBET

At any stage in the program it's possible to stop with the program or program part by typing in the command EXIT (Except when the computer is performing calculations). If, for instance, you are working in the advance die filling mode and you want to stop, then you have to enter EXIT twice, once for leaving the advance die filling mode and once for leaving the main program.

4. Possible forging analysis

4.1. Forging load prediction

After entering the geometries and other data forging analysis can be performed, but prior to the load prediction, it's necessary to divide the billet into the required number of elements. This can be done automatically by the command DIVIDE. In return this command will display the number of elements on the screen. The number of elements can be increased with the command HIGHER, or decreased with the command LOWER, which also display the number of elements on the screen.
When a satisfying level of subdivision is reached, load calculation is initiated by the command LOAD.

4.2. Flowlines prediction

Prediction of the possible metal flow is only operable after the determination of the forging load, because flowline calculation demands precalculation of the elemental velocities. In order to look at the flowlines type the command FLOWLINES or its abbreviation FLO after the calculation of the load.

4.3. Advance die filling

This feature predicts a possible metal filling inside a pair of closed dies. To use this feature enter ADVANCE or its abbreviation ADV; then the program will ask the operator for the required number of increments
(for instance 5 or 10) and after that return with the prompt GO>.
There are further possibilities inside this feature, see the enclosure about UBET commands.

4.4. Die billet interfacial pressure

In UBET the die billet interfacial pressure can be estimated at any stage after the input of billet geometrie and the modification of the relevantly defaulted parameters. To activate the Pressure Element Technique (PET), the command PRESSURE is used, after the prompt of GO>, and then the program starts asking you the information it needs for performing the necessary calculations, such as which friction model, which point, the element width, the velocity change in procents and which die is in contact with the billet at the point. After this, return information is displayed on the screen, followed by calculation of the power consumption in forging the billet for both the positive and the negative velocity change.
Finally, output results are :

forging load for a velocity change of 1.0 % on the region,
the local surface pressure, and
the contact area.

ENCLOSURE 1 COMMANDS' DESCRIPTION:

ADVANCE prompts 'ADV:'

Simulate the forging process from start to end in an incremental manner. This module has a set of sub–commands through which the output features are set. The sub–commands are:

REPORT: promts 'REP:'

this command has the following parameters:

- GO: The main program command 'report' is invoked at every increment.
- MESH: A (10x10) mesh is imposed on the billet cross section and the final grid coördinates and the strain experienced by every point area amended to the output file.
- STRAIN: The regional average strain values are directed to the report file.

DISPLAY: display current component geometry and dies position at the beginning of every step.

MESH Impose and display a (10x10) mesh with the current component geometry and dies position.

GO: parameters are set, advance dies.

SIMULATE: The process from start to end is simulated. This can only be used if the GO command is succesfully executed.

EXIT: return to main commands mode.

BILLET	prompts 'BIL:' Input/edit billet coördinates. On exit the billet volume is showed.
BOTTOM	prompts 'BOT:' Input/edit bottom die coördinates.
CAVITY	Calculate the final cavity volume, i.e. volume enclosed by the dies for a given penetration.
DATA	Display the current process variables which are used in load calculations.
DIVIDE	Create a reasonably fine subdivision and reset the approximation level to the default.
DRAW	Draw the available outlines onto a graphics terminal.
EXIT	Stops the program.
FLOWLINES	Display the predicted flowlines inside the forging.
FRICTION	prompts 'F:' Friction factor.
HIGHER	A new subdivision is generated with an approximation level which is higher than that of the current level.
LOAD	Calculate forging load.
LOWER	A new subdivision is generated with an approximation level which is lower than that of the current level.
MATERIAL	prompts 'MATERIAL:' Billet material.
MODIFY	Modify the recent subdivision for lesser number of regions, but in the current level of approximation.

MOVE	prompts 'MOVE:' Move 'TOP' or 'BOTTOM' die or 'BILLET' coördinates a distance 'd' ('+' up and '–' down). Format: MOVE: 'coördinates' 'd'.
NEW	Prepare the program for receiving a new component geometry, no changes in the process variables will take place.
PENETRATION	prompts 'P:' Total die travel for an incremental solution.
PLOT	Direct the available outlines to the graph plotter. (This command is not available yet!!)
PROCESS	prompts 'PROCESS' Type of process: normal, hot or cold.
PRESS	Starts pressure element technique.
READ	prompts 'FILE NAME:' Read in available coördinates and reference from an existent data file created by the same package.
REFERENCE	prompts 'REFERENCE:' Change reference.

REPORT The following data – if available – are directed to a file which is printed at exit:

Reference,

Billet coördinates,

Top die coördinates,

Bottom die coördinates,

Billet volume,

Number of elemental regions,

% volume change due to approximation,

Friction factor,

Die velocity,

RPM, stroke, displacement,

Material flow stress,

Forging material,

Forging temperature,

Process type,

Penetration and

Forging load.

If the command is repeated during a session only the parameters which have been changed will be appended to the current output file.

SPEED prompts 'RPM, STROKE, then DISPLACEMENT'

Estimated die velocity according to the working conditions.

STRESS prompts 'A then B'

The stress–strain/or strain rate relation is expressed by:

$STRESS = A + B \cdot \bar{\epsilon}$ (Cold forging)

$STRESS = A + B \cdot \log(\dot{\bar{\epsilon}})$ (Hot forging)

TEMPERATURE	Prompts 'T'
	Forging temperature.
TOP	Input/edit top die coördinates.
VELOCITY	Prompts 'V:'
	Die velocity.
VOLUME	Show billet volume.
WRITE	prompts 'FILE NAME:'
	Copy coördinates and reference into the specified data file.

INPUT/EDITOR QUALIFIERS:

This module is used for input and/or edit of the outlines coördinates. The input mode is considered the normal mode unless an editor qualifier is entered. This technique avoids the use of a selective file copying procedure and therefore, the input/editor pointer can go upward or downward at will. Coördinates should be entered in a sequencial manner starting from the lowermost point on the center line. The editor qualifiers have their special meaning and a qualifier is processed, only if it is proceeded by a point number. No action will be taken if the command is out of order, for example, the user requires a non–existent point. However, the editor gives an explanatory message which describes the action taken.

NOMENCLATURE:

'n' Point number.

'x' X–coördinate.

'y' Y–coördinate.

'm' Point marker,

it denotes the next billet surface neighbour with

'C' for center

'T' for top die

'B' for bottom die

'F' for flash.

INPUT MODE:

'n', 'x', 'y'	Top or bottom die coördinates.
'n', 'x', 'y', 'm'	Billet coördinates.

It is not necessary to supply the marker 'm' if the preceding point carries the same marker.

EDITOR QUALIFIERS

D,'n'	Delete point number 'n'
X,'n','x'	Replace the X–coördinate of point 'n' by 'x'.
Y,'n','y'	Replace the Y–coördinate of point 'n' by 'y'.
M,'n','m'	Replace the marker of point 'n' by 'm'.
M,'n'<CR> <CR>	Delete the marker of point 'n'.
L,'n1','n2'	List points in the range n1–n2.
S,'n','x','y','m'	Shift points, starting from 'n', one step forward and replace the coördinates and marker of point 'n' by 'x', 'y', 'm'.
Exit	Exits editor, program returns to main menu.

MATERIAL LIBRARY

EN8	(800–1300 degree 'C)
HCS	High Carbon Steel (900–1200 degree 'C)
LCS	Low Carbon Steel (900–1200 degree 'C)
MCS	Medium Carbon Steel (900–1200 degree 'C)

ENCLOSURE 2. EXAMPLE OF INPUT

In this enclosure a full description is given of the input of one specific component, shown below.

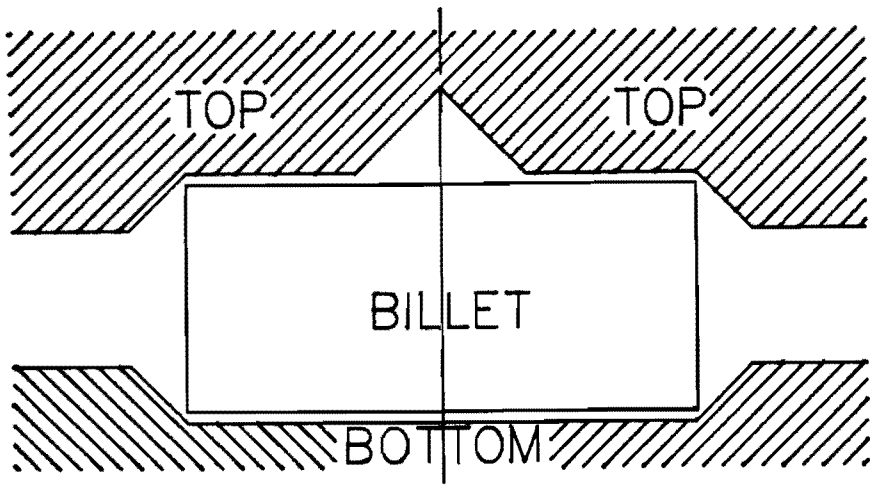

figure 1: component

First thing to do is to start the UBET program.

Change drive and directory to those where the UBET program is stored, type :

1 C: [program is stored on drive C]

CD\UBETEXIS [in the directory UBETEXIS]

Start the program itself, type :

2 ADV [the forward solution of UBET is started]

Secondly, the input has to be given.

The program prompts with: PLOTTER CONNECTED <Y> <N> ?

3 So type N <> [No plotter connected]

The program will respond with REFERENCE : , for instance type :

4 THATCHER [if the operator's name is Thatcher]

The program will respond with a prompt of GO

The geometries of the dies and the billet have to be supplied, type :

5 BIL [inputting billet geometry]

A prompt of BIL wil follow. Now the program is ready for input, type :

6	1	0.0	0.0	C	[pointno 1, x and y–coordinate, centerline marker]
	2	0.0	40.0	F	[at this position the flash starts]
	3	15.0	40.0	T	[first point in contact with the top die]
	4	30.0	40.0	F	[again the start of free surface]
	5	30.0	0.0	B	[first point in contact with the bottom die]
	E	[Exit of billet input mode]			

Now the billet geometry is defined, next will be the bottom die.

Type :

7 BOT [inputting bottom die geometry]

A prompt of BOT will appear on the screen, type :

8	1	0.0	0.0	[respectively pointnumber, x and y coordinate]
	2	30.0	0.0	
	3	40.0	10.0	
	4	60.0	10.0	
	E	[Exit of bottom die input mode]		

After the bottom die geometry, the top die geometry has to be defined, type :

9 TOP [inputting top die geometry]

A prompt of TOP will appear on the screen, type

10	1	0.0	55.0	[respectively pointnumber, x and y coordinate]
	2	15	40.0	
	3	30.0	40.0	
	4	40.0	30.0	
	5	60	30.0	
	E	[Exit of top die input mode]		

The component is fully defined and can be saved by giving the command :

11 WRITE COMPONENT [COMPONENT is the name of the component]

When the geometry input is given, the remaining parameters of the process, like the penetration, yield stress etc. have to be set.

For defining the penetration, type :

12 PEN 10 [10 mm of penetration]

In order to view the process parameters, type :

13 DATA [view default data]

The program will respond with the default values of the friction factor, stress, die–velocity, type of process, material and process temperature.

For changing one of the above, type the first three letters of the item, followed by the new value, for instance :

14 PRO COLD [set process to cold]

15 VEL 10 [set die velocity to 10 mm/sec]

When the data are correct then the UBET analysis can start, type :

(Here is chosen to use the UBET advance die filling)

16 ADV

The program responds with REQUIRED NUMBER OF INCREMENTS , type :

17 10 [process will be simulated in 10 steps]

followed by :

18 GO [start calculations]

The program prompts LEVEL OF SUBDIVISION OK (Y/N) ? , and shows forging plus subdivision into elements on the screen. Type :

19 Y [level of subdivision ok] or

19 N [level of subdivision not ok], the program will generate more elements.

After each step the program prompts : CONTINUE (Y/N) ? , type :

20 Y [program will continue]

20 N [program will abort advance mode]

As a result of each step the program will plot the new geometry on the screen, together with the resulting die force. In addition for each step an output file is written (INC1, INC2, INC3 etc).

When the operator wants to run the analysis again, with different data, it's not necessary to define the geometry of billet and dies again. With the READ command an already existing file can be read. Type :

21 READ COMPONENT [COMPONENT is the name of the component]

And then define the new data. Furthermore it's possible to edit an existing file, in case a mistake was made while putting in the geometry of for instance the bottom die.

22 BOT

23 Y 2 56 [Replace Y coordinate of point 2 with 56]

All UBET commands are described in enclosure 1.

ENCLOSURE 3.1. UBETEXIS

The necessary source files for compiling and linking of the existing UBET are:

AAI	FOR	C	FOR	C1	FOR
CHAMOD	FOR	D	FOR	D1	FOR
D2	FOR	D2R	FOR	I1	FOR
I2	FOR	I3	FOR	I4	FOR
I5	FOR	IO	FOR	IO1	FOR
IO2	FOR	IO3	FOR	IOR	FOR
L	FOR	L1	FOR	L2	FOR
L3	FOR	L4	FOR	L5	FOR
LR	FOR	M	FOR	P	FOR
P1	FOR	P2	FOR	P3	FOR
PLOT	FOR	Q	FOR	Q1	FOR
Q2	FOR	Q3	FOR	Q4	FOR
Q5	FOR	Q6	FOR	QMAX	FOR
REV	FOR	S	FOR	S1	FOR
S1R	FOR	S2	FOR	S2R	FOR
S3	FOR	S3R	FOR	S4	FOR
S4R	FOR	S5	FOR	S5R	FOR
S6	FOR	S6R	FOR	S7	FOR
S7R	FOR	SR	FOR	T	FOR
T1	FOR	T2	FOR	T3	FOR
T4	FOR	T5	FOR	T6	FOR
T7	FOR	W	FOR	X	FOR
X1	FOR				

Except these fortran source files, you also need the following object files:

CLEAR	OBJ	PLOTPE	OBJ
PLOT	OBJ	PTABLE1	OBJ
PEPLOT	OBJ		

And of course, you need the libraries:

PROFORT	LIB
PFGKS	LIB
GKS	LIB

For automatic compiling, linking and installing the existing UBET, the following bat file is used.

EXISINST.BAT:

```
CLS
ECHO OFF
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO    This batch file is meant to install, compile and link the
ECHO    existing UBET program.
ECHO
ECHO    So without use of the SONG BIN optimiser
ECHO
ECHO
ECHO    GO ON      : strike any key,
ECHO    STOP       : press control-C
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
PAUSE
ECHO ON
CLS
C:
CD\
MD UBETEXIS
CD UBETEXIS
MD FOR
CD FOR
MD FORTRANF
MD OBJ
CLS
ECHO OFF
ECHO
ECHO
ECHO
ECHO        INSERT DISC 1, AND STRIKE ANY KEY
ECHO
ECHO
ECHO
PAUSE
ECHO ON
```

```
COPY A:\LNK\*.* C:\UBETEXIS\FOR\OBJ
COPY A:\FOR\FORTRANF\*.FOR C:\UBETEXIS\FOR\FORTRANF
ECHO OFF
CLS
ECHO
ECHO
ECHO
ECHO          INSERT DISC 2, AND STRIKE ANY KEY.
ECHO
ECHO
ECHO
PAUSE
ECHO ON
COPY A:\FOR\FORTRANF\*.FOR C:\UBETEXIS\FOR\FORTRANF
COPY A:\FOR\OBJ\*.OBJ C:\UBETEXIS\FOR\OBJ
COPY A:\DATAFLS\*.* C:\UBETEXIS
ECHO OFF
CLS
ECHO
ECHO
ECHO          COMPUTER NOW STARTS WITH COMPILING AND LINKING.
ECHO          ESTIMATED TIME ON A PC-XT  4 HOURS
ECHO
ECHO          IN THE MEAN TIME YOU DON'T HAVE TO DO ANYTHING.
ECHO
ECHO
PAUSE
ECHO ON
path C:\UBETTOOL;
CD\
CD\UBETEXIS\FOR\FORTRANF
PROFORT AAI
PROFORT C
PROFORT C1
PROFORT CHAMOD
PROFORT D
PROFORT D1
PROFORT D2
PROFORT D2R
PROFORT I1
PROFORT I2
PROFORT I3
PROFORT I4
PROFORT I5
COPY *.OBJ C:\UBETEXIS\FOR\OBJ
DEL  *.OBJ
PROFORT IO
PROFORT IO1
PROFORT IO2
PROFORT IO3
PROFORT IOR
PROFORT L
PROFORT L1
```

```
PROFORT L2
PROFORT L3
PROFORT L4
PROFORT L5
PROFORT LR
COPY *.OBJ C:\UBETEXIS\FOR\OBJ
DEL  *.OBJ
PROFORT M
PROFORT P
PROFORT P1
PROFORT P2
PROFORT P3
PROFORT PLOT
PROFORT Q
PROFORT Q1
PROFORT Q2
PROFORT Q3
PROFORT Q4
PROFORT Q5
PROFORT Q6
PROFORT QMAX
COPY *.OBJ C:\UBETEXIS\FOR\OBJ
DEL  *.OBJ
PROFORT REV
PROFORT S
PROFORT S1
PROFORT S1R
PROFORT S2
PROFORT S2R
PROFORT S3
PROFORT S3R
PROFORT S4
PROFORT S4R
PROFORT S5
PROFORT S5R
COPY *.OBJ C:\UBETEXIS\FOR\OBJ
DEL  *.OBJ
PROFORT S6
PROFORT S6R
PROFORT S7
PROFORT S7R
PROFORT SR
PROFORT T
PROFORT T1
PROFORT T2
PROFORT T3
PROFORT T4
PROFORT T5
PROFORT T6
PROFORT T7
PROFORT W
PROFORT X
PROFORT X1
```

```
COPY *.OBJ C:\UBETEXIS\FOR\OBJ
DEL  *.OBJ
CD..
CD OBJ
SET LIB=C:\UBETTOOL\LIB
RENAME AAI.OBJ ADVF.OBJ
LINK @FMAKE1.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
LINK @RMAKE.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
RENAME ADVF.OBJ ADVM.OBJ
LINK @FMAKE.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
COPY *.EXE C:\UBETEXIS
DEL *.EXE
CD\
CLS
ECHO OFF
ECHO
ECHO
ECHO          COMPILING AND LINKING FINISHED.
ECHO          EXECUTABLE EXE FILES : REV, ADVM, ADVF.
ECHO
ECHO          THIS INSTALLATION HAS CHANGED THE PATH,
ECHO          SO YOU BETTER REBOOT YOUR SYSTEM.
ECHO
ECHO
ECHO ON
PAUSE
```

In this bat file link files are used for linking the object files into an executable program.

The file fmake.lnk contains the following:

```
ADVM+C+C1+CHAMOD+CLEAR+D+D1+D2+I1+I2+I3+I4+I5+IO+IO1+IO2+IO3+
L+L1+L2+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+Q+Q1+Q2+Q3+Q4+Q5+
Q6+QMAX+S+S1+S2+S3+S4+S5+S6+S7+T+T1+T2+T3+T4+T5+T6+T7+X+X1+
W+PEPLOT+PLOTPE+PTABLE1
```

The file fmake1.lnk contains the following:

```
ADVF+C+C1+CHAMOD+CLEAR+D+D1+D2+I1+I2+I3+I4+I5+IO+IO1+IO2+
IO3+L+L1+L2+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+Q+Q1+Q2+Q3+Q4+Q5+
Q6+QMAX+S+S1+S2+S3+S4+S5+S6+S7+T+T1+T2+T3+T4+T5+T6+T7+X+X1+W
```

The file rmake.lnk contains the following:

```
REV+C+C1+CHAMOD+CLEAR+D+D1+D2R+I1+I2+I3+I4+I5+IOR+IO1+IO2+
IO3+LR+L1+L2+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+
SR+S1R+S2R+S3R+S4R+S5R+S6R+S7R+T+T1+T2+T3+T4+T5+T6+T7+X+X1
```

ENCLOSURE 3.2. UBETSB

The necessary source files for compiling and linking of the Song Bin UBET are:

AAI	FOR	C	FOR	C1	FOR
CHAMOD	FOR	D	FOR	D1	FOR
D2	FOR	D2R	FOR	I1	FOR
I2	FOR	I3	FOR	I4	FOR
I5	FOR	IO	FOR	IO1	FOR
IO2	FOR	IO3	FOR	IOR	FOR
L	FOR	L1SB	FOR	L2	FOR
L2SB	FOR	L3	FOR	L4	FOR
L5	FOR	LR	FOR	M	FOR
P	FOR	P1	FOR	P2	FOR
P3	FOR	PLOT	FOR	Q	FOR
Q1	FOR	Q2	FOR	Q3	FOR
Q4	FOR	Q5	FOR	Q6	FOR
QMAX	FOR	REV	FOR	S	FOR
S1	FOR	S1R	FOR	S2	FOR
S2R	FOR	S3	FOR	S3R	FOR
S4	FOR	S4R	FOR	S5	FOR
S5R	FOR	S6	FOR	S6R	FOR
S7	FOR	S7R	FOR	SR	FOR
T	FOR	T1	FOR	T2	FOR
T3	FOR	T4	FOR	T5	FOR
T6	FOR	T7	FOR	W	FOR
X	FOR	X1	FOR		

Except these fortran source files, you also need the following object files:

CLEAR	OBJ	PLOTPE	OBJ
PLOT	OBJ	PTABLE1	OBJ
PEPLOT	OBJ		

And of course, you need the libraries:

PROFORT	LIB
PFGKS	LIB
GKS	LIB

For automatic compiling, linking and installing the Song Bin UBET, the following bat file is used.

SBINST.BAT

```
CLS
ECHO OFF
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO    This batch file is meant to install, compile and link the
ECHO    Song Bin UBET program.
ECHO
ECHO    So with use of the SONG BIN optimiser
ECHO
ECHO
ECHO    GO ON      : strike any key,
ECHO    STOP       : press control-C
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
ECHO
PAUSE
ECHO ON
CLS
C:
CD\
MD UBETSB
CD UBETSB
MD FOR
CD FOR
MD FORTRANF
MD OBJ
CLS
ECHO OFF
ECHO
ECHO
ECHO
ECHO        INSERT DISC 1, AND STRIKE ANY KEY.
ECHO
ECHO
ECHO
PAUSE
ECHO ON
```

```
COPY A:\LNK\*.* C:\UBETSB\FOR\OBJ
COPY A:\FOR\FORTRANF\*.FOR C:\UBETSB\FOR\FORTRANF
ECHO OFF
CLS
ECHO
ECHO
ECHO
ECHO          INSERT DISC 2, AND STRIKE ANY KEY.
ECHO
ECHO
ECHO
PAUSE
ECHO ON
COPY A:\FOR\FORTRANF\*.FOR C:\UBETSB\FOR\FORTRANF
COPY A:\FOR\OBJ\*.OBJ C:\UBETSB\FOR\OBJ
COPY A:\DATAFLS\*.* C:\UBETSB
ECHO OFF
CLS
ECHO
ECHO
ECHO          COMPUTER NOW STARTS WITH COMPILING AND LINKING.
ECHO          ESTIMATED TIME ON A PC-XT  4 HOURS
ECHO
ECHO          IN THE MEAN TIME YOU DON'T HAVE TO DO ANYTHING.
ECHO
ECHO
PAUSE
ECHO ON
path C:\UBETTOOL;
CD\
CD\UBETSB\FOR\FORTRANF
PROFORT AAI
PROFORT C
PROFORT C1
PROFORT CHAMOD
PROFORT D
PROFORT D1
PROFORT D2
PROFORT D2R
PROFORT I1
PROFORT I2
PROFORT I3
PROFORT I4
PROFORT I5
PROFORT IO
PROFORT IO1
COPY *.OBJ C:\UBETSB\FOR\OBJ
DEL *.OBJ
PROFORT IO2
PROFORT IO3
PROFORT IOR
PROFORT L
```

```
PROFORT L1SB
PROFORT L2SB
PROFORT L2
PROFORT L3
PROFORT L4
PROFORT L5
PROFORT LR
PROFORT M
PROFORT P
PROFORT P1
PROFORT P2
PROFORT P3
COPY *.OBJ C:\UBETSB\FOR\OBJ
DEL *.OBJ
PROFORT PLOT
PROFORT Q
PROFORT Q1
PROFORT Q2
PROFORT Q3
PROFORT Q4
PROFORT Q5
PROFORT Q6
PROFORT QMAX
PROFORT REV
PROFORT S
PROFORT S1
PROFORT S1R
PROFORT S2
PROFORT S2R
PROFORT S3
PROFORT S3R
COPY *.OBJ C:\UBETSB\FOR\OBJ
DEL *.OBJ
PROFORT S4
PROFORT S4R
PROFORT S5
PROFORT S5R
PROFORT S6
PROFORT S6R
PROFORT S7
PROFORT S7R
PROFORT SR
PROFORT T
PROFORT T1
PROFORT T2
PROFORT T3
PROFORT T4
PROFORT T5
PROFORT T6
PROFORT T7
PROFORT W
PROFORT X
PROFORT X1
```

```
COPY *.OBJ C:\UBETSB\FOR\OBJ
DEL *.OBJ
CD..
CD OBJ
SET LIB=C:\UBETTOOL\LIB
RENAME AAI.OBJ ADVF.OBJ
LINK @FMAKE1.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
LINK @RMAKE.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
RENAME ADVF.OBJ ADVM.OBJ
LINK @FMAKE.LNK ,,, pfgks gks profort /se:1024 /stack:20000;
COPY *.EXE C:\UBETSB
DEL *.EXE
CD\
CLS
ECHO OFF
ECHO
ECHO
ECHO          COMPILING AND LINKING FINISHED.
ECHO          EXECUTABLE EXE FILES : REV, ADVM, ADVF.
ECHO
ECHO          THIS INSTALLATION HAS CHANGED THE PATH,
ECHO          SO YOU BETTER REBOOT YOUR SYSTEM.
ECHO
ECHO
ECHO ON
PAUSE
```

In this batfile link files are used for linking the object files into an executable program.

The file rmake.lnk contains the following:

```
REV+C+C1+CHAMOD+CLEAR+D+D1+D2R+I1+I2+I3+I4+I5+IOR+IO1+IO2+
IO3+LR+L1SB+L2+L2SB+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+
SR+S1R+S2R+S3R+S4R+S5R+S6R+S7R+T+T1+T2+T3+T4+T5+T6+T7+X+X1
```

The file fmake.lnk contains the following:

```
ADVM+C+C1+CHAMOD+CLEAR+D+D1+D2+I1+I2+I3+I4+I5+IO+IO1+IO2+
IO3+L+L1SB+L2+L2SB+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+Q+Q1+Q2+
Q3+Q4+Q5+Q6+QMAX+S+S1+S2+S3+S4+S5+S6+S7+T+T1+T2+T3+T4+T5+T6+
T7+X+X1+W+PEPLOT+PLOTPE+PTABLE1
```

The file fmake1.lnk contains the following:

```
ADVF+C+C1+CHAMOD+CLEAR+D+D1+D2+I1+I2+I3+I4+I5+IO+IO1+IO2+
IO3+L+L1SB+L2+L2SB+L3+L4+L5+M+P+P1+P2+P3+PFGKS+PLOT+Q+Q1+Q2+
Q3+Q4+Q5+Q6+QMAX+S+S1+S2+S3+S4+S5+S6+S7+T+T1+T2+T3+T4+T5+T6+
T7+X+X1+W
```